kcFED Technical Working Papers

The Road to Cyberinfrastructure at the Federal Reserve Bank of Kansas City

BJ Lougee, Tim Morley, and Mark Watson

April 2018
TWP 18-02

Federal Reserve Bank of Kansas City

The Road to Cyberinfrastructure at the Federal Reserve Bank of Kansas City

By BJ Lougee, Tim Morley, and Mark Watson[*]

The Federal Reserve Bank of Kansas City's (FRBKC) Center for the Advancement of Data and Research in Economics (CADRE) has transformed computing, and in particular High Performance Computing (HPC), for our researchers. With the added computational complexity of new economic models and the increase in the amount of data that our economic researchers use, CADRE needed to develop an environment where we could facilitate better research accommodating these new factors. Now more than ever, economic research requires an environment that is flexible and allows almost any type of workload using almost any amount of data.

The computational demands of current research might rely on large distributed memory jobs (where each processor has its own memory), shared memory jobs (where there is a single memory space used by all processors), or high throughput computing (efficient execution of large number of independent sequential jobs on many different computing resources). Because different researchers have different levels of familiarity with these different techniques, a successful environment requires a low barrier of entry for non-traditional HPC researchers. The HPC infrastructure must be flexible enough to allow researchers who do not have HPC experience to feel comfortable while still allowing advanced researchers access to familiar tools. In addition, new options for working with large data sets require robust data storage and simple access from the flexible computing

[*] BJ Lougee, Computer Scientist and HPC Engineer, BJ.Lougee@kc.frb.org; Tim Morley, Senior Software Engineer, timothy.r.morley@kc.frb.org; Mark Watson, Vice President, mark.a.watson@kc.frb.org Center for the Advancement of Data and Research in Economics, Research Division, Federal Reserve Bank of Kansas City. The views expressed here are those of the author and do not necessarily reflect those of the Federal Reserve Bank of Kansas City, or the Federal Reserve System.

This paper could not have been written without the effort of several people. I would like to thank Chris Stackpole for his input onto how the COW environment was implemented and worked. I would like to thank Brad Praiswater for his work on the CADRE Cyberinfrastructure diagram. Last but most importantly, I would like to give a huge thank you to San Cannon. Without San this paper would not have happened. Her tireless feedback was invaluable.

environment. Both users and support staff need to be comfortable working in the environment with the tools provided. It can be a tall order to deliver all of these things.

To meet this order, we focused on three main areas. First, we updated our computing infrastructure to an HPC environment, improving both the computational capabilities and the supporting infrastructure. Second, we developed a data warehouse for large complex datasets that traditional database systems would not be able to handle. Finally, we worked to hire and train personnel to help facilitate the transition to the new environment for our researchers. The result is a well received, flexible cyberinfrastructure that provides access to and support for computation, data, and training.

Background

Originally economic research at FRB KC was done on individual desktop computers or UNIX servers for those researchers who knew how to access and use these expensive proprietary machines. We made the decision in 2002 to change from individual UNIX servers to a Linux Cluster of Workstations (COW) environment, a type of distributed computing where computers are loosely coupled together to provide more compute resources and parallel computing. The COW environment has multiple computational nodes which are not dedicated to one particular user but are typically shared by many users with various different workloads.

We also installed a scheduler in the environment. The scheduler was intended to help manage the competition for computing resources, but it was rarely used and use of it was not enforced, resulting in huge load imbalances. It was the computing equivalent of a demolition derby: researchers would consistently log into their favorite node regardless of the current usage. At any one time 80% of the researchers would be using one or two nodes while the rest were idle. If one user overloaded the system, they could bring down the compute node. All of the researchers would lose their work so they would need to start all over again. They would pick a different node, and the same process would repeat.

While an improvement over standalone workstations, COW environments are typically not good for long running computational programs due to the diversity of programs running on the nodes at any one time. That said, the COW environment allowed us to scale out and incrementally change or improve the environment. In addition, researchers could utilize open source software which allowed researchers to take advantages of more compute power than costly licensed software which can artificially limit the amount of computation power available. By 2008 many researchers found that the COW environment was the only hardware available that could handle their new workloads, even if they were previously unfamiliar with the environment. Because the networked workstations had a similar software base to the standard desktops, researchers had access to server hardware with a familiar desktop interface thereby increasing their ability to easily use the increased computing power. This architecture provided an introduction for our researchers to move a step beyond doing computational work on their workstations.

Eventually the researchers learned how to work within the COW system but the size and complexity of new data sources and the computational approaches that researchers were exploring showed that the derby was not sustainable. In the challenging economic environment of the early 2000s, our traditional database systems were challenged with high input-output (IO) and low central processing unit (CPU) utilization where most of the run time was spent waiting on data from files or from the database and less time was spent processing the actual requests. In addition, the size and complexity of the new data sources made it challenging to do data preparation, exploratory data analysis, or modeling. In order to continue to move economic research forward, we needed a new solution.

Creating a Cyberinfrastructure

After ten years with our COW architecture, we started to move to an HPC environment where we could bring more advanced computing practices to our researchers. This move presented challenges given the knowledge of our user base. In the HPC environment, the researchers required better computational technical knowledge and experience, familiarity with a wide range of specialized software packages, and experience with various levels of the operating system in order effectively use traditional HPC resources. To address this, we have developed not just an HPC environment but a robust cyberinfrastructure.

The term "cyberinfrastructure" was first coined by the National Science Foundation in 2003 (Atkins et. al., 2003). A widely used definition is given by the University of Indiana: Knowledge Base: "Consists of computing systems, data storage systems, advanced instruments and data repositories, visualization environments, and people all linked by high speed networks to make possible scholarly innovation and discoveries not otherwise possible" (2017). Alternatively, Educause states that cyberinfrastructure is a ". . . component of information technology focused on distributed computing, data, and communications technology. Hardware and software systems are quickly being developed and implemented to build the research communities along with collaborative tools to knit these user communities together"(2018).

CADRE specializes in catering to economic researchers with a cyberinfrastructure that is designed specifically with them in mind. Our HPC infrastructure is a variation of a traditional cluster. With a traditional cluster, a user can only access it by connecting with a secure shell through a login node that only runs jobs by batch submission. Instead, we have three different types of nodes: compute nodes, interactive nodes, and command line nodes where light work can be done including launching batch submission jobs.

In our environment, the compute nodes consist of 16 core processors at 3.20GHz with 128 GB of memory. These nodes allow researchers to run distributed memory jobs with Message Passing Interface (MPI) or, shared memory jobs like OpenMP. Interested researchers can also utilize high throughput computing by doing parameter sweeps with single core jobs instead of running serially, which drastically cut down the time it takes to get their results.

We also have several different types of interactive nodes. The first of these, the sandbox desktops, gives researchers an environment to do development and prototyping of code. These nodes allow an entry point into the HPC environment that can help transition researchers to more traditional HPC techniques. The sandbox desktop nodes are typically recycled hardware from previous cluster iterations. The current sandbox desktop nodes are 12 core processors at 2.67GHz with 96GB of memory. Besides the desktop nodes, we have more powerful interactive nodes that typically have more memory where researchers can have interactive sessions to develop code on large datasets. These more powerful interactive nodes have the same hardware configuration as the compute nodes but with two or four times the amount of memory. A third type of interactive node, the accelerator nodes, have various graphical processing units (GPGPUs) and many-core accelerators. All of the interactive nodes have a Linux desktop where researchers will feel more comfortable working within their chosen languages and software in an interactive session. The baseline hardware configuration for these nodes are 24 core processors at 2.30GHz with 128GB of memory.

Finally, we have a few command line nodes which are what would be considered traditional login nodes in other environments. These are not pure login nodes in our environment, however, since we have dedicated login nodes that handle a web portal used by most researchers to launch their jobs. Researchers who have had experience with HPC resources and only need command line access can connect to these nodes to launch batch jobs on our compute nodes. As the command line nodes aren't strictly login nodes, they are also available for light computational work, providing yet another avenue for accessing our environment. The command login nodes are 8 core processors at 2.80GHz with 108GB of memory.

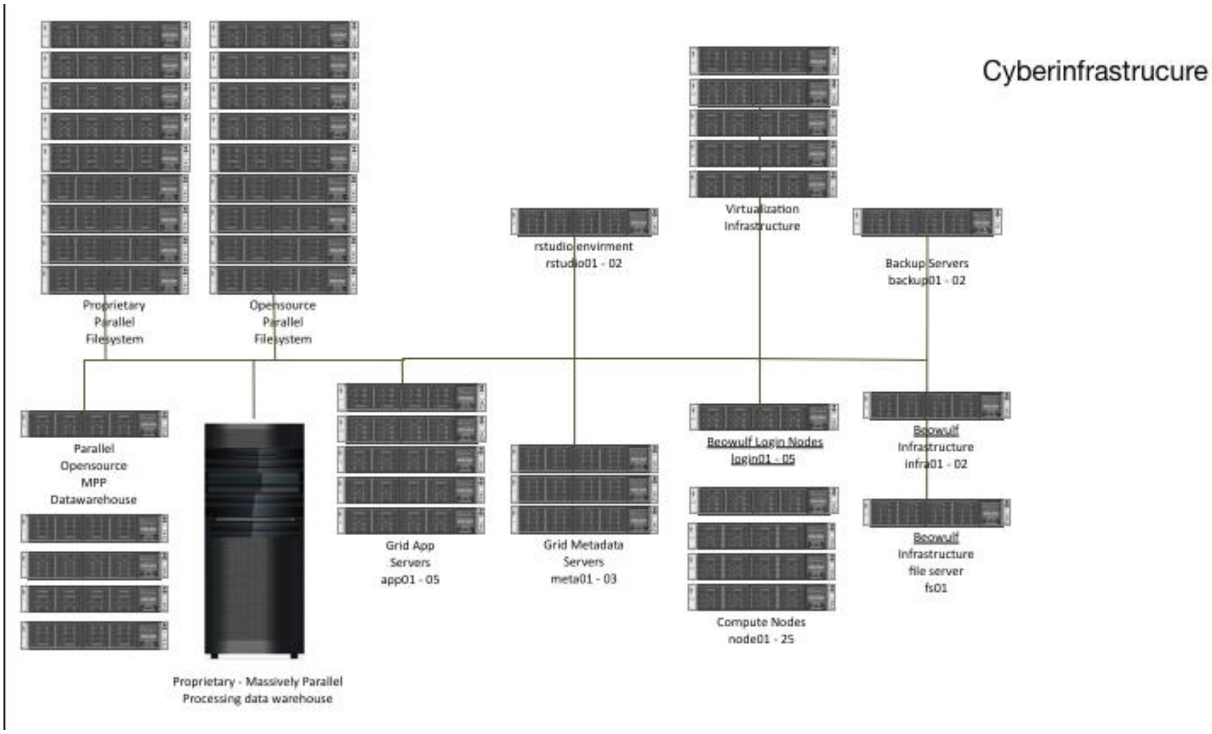

Figure 1: CADRE cyberinfrastructure

Updated Data Warehousing

The improved computing platform is just one component of the new cyberinfrastructure. The data warehousing group began work in late 2008 to provide an environment for researchers to work with data stored in a relational database management system (RDBMS). Researchers wanted the ability to write their own queries directly against the raw data and they needed to have the environment up and running as soon as possible. We initially overlooked a key item in the planning: the amount of data with which researchers would be working at any one time. At the time, nearly every RDBMS worked off the design that a single query used one CPU in a system no matter how many CPUs or CPU cores were available. This architecture struggled to handle the load of users submitting queries to view or extract data as well as the load of updating data at the same time. For example, one job to create user defined variables from stored data took 48 hours to process. Such lengthy runtime was not acceptable for users so a new system was required.

By the end of 2008 we discovered a RDBMS system for data warehousing that was using a massively parallel processing (MPP) architecture. The new MPP system chosen shared a code base with PostgreSQL, a technology with which our staff was already familiar, thereby reducing the costs of managing and configuring the new system. In the fall of 2009, we began a proof of concept with the new system with a variety of sample queries. Initial benchmarking of the MPP system against our PostgreSQL configuration was jaw dropping. Sample queries were running hundreds of times faster in the MPP system than our existing PostgreSQL system. An MPP system has multiple computers (segment hosts) dividing up the data into small sections with each processing core in the environment working on a small subset of the data. For example, a small MPP system may include the master host that handles incoming requests and 4 segment hosts that handle data storage and processing. If a PostgreSQL system had 8 processing cores, one core would handle running any single query. In an MPP architecture, if each segment host had 8 processing cores, then 4 cores in each segment host would be used to process queries. The other 4 would be used for backup work. This means that 16 CPU cores would be used for processing a query instead of the single core in a traditional data warehouse system. In addition, each core would handle 1/16 of the amount of data that the single processing query had to manage.

In Spring 2009, the first iteration of our MPP system was introduced into the environment. Again speed gains were simply astounding. One of the first jobs to be moved into the environment was a job that calculated aggregate statistics across states. The previous process pulled data out of the PostgreSQL system and then used analytics software to create the aggregated data and resulting output, a process which routinely took more than 3 hours to complete. In the new system, the two steps could be combined so that a single query produced the same results in 15 seconds. The data loading job run by CADRE staff that originally required 48 hours to create new variables would now run in 90 minutes. Users and staff alike were very happy with these results.

As is true for the use of HPC technologies, there are varying levels of technical knowledge for database access within that population. We strive to provide as much flexibility as possible to access the data based on the users' preferences and technical abilities. Users are able to query data from the HPC environment, dedicated analytics software servers, or using query tools available on their laptops or desktop systems. While writing queries will provide the most flexible experience for individuals, some do not have the desire to learn to write complex SQL queries. For those users, we have provided a web based query tool that makes writing queries as simple as pointing and clicking on variable names. We also provide users the flexibility to edit their pre-built queries and the ability to upload reference files to aide in returning a targeted set of observations. For users with higher levels of technical experience we also provide an in database analytic system that allows users to push computational work directly into the MPP system. This can dramatically reduce the time to complete computationally intensive tasks. Regardless of the preferred access method, the work is still done securely. All remote connections are encrypted to provide security for users. Authentication to the database uses network passwords to provide an easy experience for users.

Cyberinfrastructure Personnel

To make the move to the new cyberinfrastructure, we needed personnel to manage the transition and run the new environment. As not all our researchers are well versed in all aspects of the environment, training is an essential part of the transition and continues to be an ongoing process with all of our users. Critical to the success of the transition, and acceptance of the environment, are our front-line personnel. They are responsible for interactions with the users and give all new researchers a one-on-one onboarding experience. These sessions instruct users in the various ways to access and use our cyberinfrastructure, give us insight into their software needs and their level of experience in using an HPC environment, and allows us to understand where we need to offer additional training. Once users have the basic information on how to access the environment, our cyberinfrastructure personnel also help them with more sophisticated tasks such as learning the job scheduler, writing batch submission scripts, constructing efficient database queries, using accelerators, and various other skills.

In addition, CADRE's HPC Engineers and Software Engineers also routinely work with researchers on code optimization. The user may request assistance directly or CADRE's HPC engineers may suggest improvements when they see potential problems with jobs that are running in the environment. Many times, the engineers can suggest small changes that use the scheduler more effectively, limit resource clashes, reduce the load on the data warehouse, and lessen the effect one researcher's work may have on other users in our HPC Environment. In addition to code, CADRE staff work to help reduce the burden and entry costs for researchers new to the environment: we have personnel embedded with the researchers who help with many aspects of reducing the entry cost to include: training, creating custom forms that are tailored to each researcher's specific workflow, and a more tailored environment for research with a fast turnaround on implementing new technology and ideas.

Choices and Benefits

Conscious thought went into transforming the infrastructure to ensure it could scale to meet the growing needs of our researchers. To improve efficiency, all of our resources in our environment are tied in with our job scheduler. This helps us avoid the demolition derby of the COW environment. We use an open source scheduler with different queues to partition out the various hardware based the type of node best suited to the job. The nodes are connected by 10Gb Ethernet which provides the fastest speeds we think suitable for the current or planned usage of our economists.

Because our users have various levels of computational experience, we provide as many ways as possible for them to access our HPC and data warehouse environments. We provide software that allows access to our clusters through a web interface that can be launched on their workstations. Inside the web interface, users have several choices. They can use a virtual desktop through a virtual network computing (VNC) session or they can access an application that is remotely forwarded to their local workstations. We provide a custom webpage that allows our researchers to upload files, specify program arguments, and automatically compile and submit a job to the job scheduler. For command-line oriented users, we provide a path to a command-line node in our HPC environment where they can submit batch jobs and/or run light computational work. Finally, we can allow our researches to remotely submit jobs for some software to our job scheduler from their workstations.

These infrastructure changes have provided benefits to both the users and CADRE staff. The users get improved performance as well as new services. For example, we have been able to introduce containerization to our researchers. The National Science Foundation (NSF) is recommending sustainability and reproducibility for research and containerization provides the ability to capture the entire research environment including code, software, and operating system configuration thereby guaranteeing continued repeatability of research regardless of changes to the cluster or other technology advances. Containerization also helps CADRE staff manage a variety of software releases needed for a wide range of projects while maintaining some standard offerings in an enterprise Linux environment. Researchers can now access legacy code running old software releases as well as the most cutting edge versions that may not yet be ready to be integrated into our environment.

The staff benefits from simplified software maintenance and provisioning. The wide range of specialized software packages was difficult to maintain in the old environment. Now all software is stored on our parallel filesystem so that it can be used throughout our HPC environment. This allows us to be able to run the software on demand without having to partition out compute nodes for specific software. This Software Defined Storage system has led to better hardware utilization. It also allows us to expand our storage as necessary at a reasonable cost with minimal downtime and with reasonable performance.

Feedback

For the most part, feedback on the new environment has been positive. Researchers appreciate the variety of ways we allow access to the environment and the separation of the interactive and batch nodes has been well received. Researchers who use the compute nodes for their jobs appreciate the minimalistic configuration of the compute nodes since they can squeeze more computation out of the existing hardware. In addition, the accelerator nodes are also popular with researchers looking at new ways to get speed improvements and the GPGPUs are particularly popular with our data science team. The new data warehouse has also garnered accolades as users like the speed and performance that was gained by moving to a MPP system. The ability to scale the performance with the size of the data has allowed our researchers to use more data with their models. In order to stay connected to changes in the research areas and continue to best serve their needs, we are planning to institute a regular survey to collect data to add to the feedback and anecdotal evidence we have so far.

In addition to hearing from our users, we are interested in connecting with other economic researchers who do work in an HPC environment. To that end, we sponsor an annual workshop on Economic Research in High Performance Computing Environments to help showcase HPC techniques that enable and advance economic research. Researchers present their work emphasizing the use of HPC environments, usually our environment but not always, and they discuss new techniques or the variety of approaches they may have taken to solve difficult research problems. The workshop brings technologists, academics, and researchers together to help foster relationships and make the connections necessary to help push the economics field forward. The workshop also provides insight for our staff on new systems and techniques to improve our environment and allows us to showcase what we have accomplished in our move from traditional HPC resources to our broader offerings that also include data-intensive research, in-database analytics, data hosting, machine learning and data science.

Future Direction

As research needs will never be static, we know that more work will always be needed. We know that we need additional user education as well as a better transition plan for researchers who are just becoming familiar with cyberinfrastructure, using HPC resources, and utilizing more advanced data resources. One challenge is managing expectations: the notion of possibly being in a queue for our shared resources can be difficult for researchers who are accustomed to running code on their laptop and having all the resources available on demand, even if they are much more limited resources. We also need to assist researchers with appropriate resource assignment so that interactive nodes are used for work that is truly interactive and keep nodes from sitting idle, thus preventing others from using that resource. We have found that a vast majority of jobs users believe to be interactive actually run more efficiently in batch mode and are working with researchers to make suitable changes. In addition, the movement in economics for research reproducibility encourages us to find ways to make it easier for researchers to containerize their code and archive it. We are exploring ways to allow researchers to utilize containerization solutions without compromising security while still making it as seamless as possible.

Our successful MPP data warehouse implementation can also benefit from improvements in speed and performance. For example, some users issue complicated queries on large datasets that still take days to get results. In addition, the data warehouse can be further optimized to make available more common variables to improve retrieval time. Finally, we'll be investigating solutions for unstructured ad-hoc databases for which we do not currently have an optimal platform. We are looking at solutions that will allow researchers to work with different types of data that has the same or better performance as the data we currently house.

We have created a unique environment tailored specifically with economic researchers in mind. Having dedicated cyberinfrastructure staff who are able to effectively interact with the users has been a critical part of the technology rollout. The improvement of our computing and data environment has made it more user friendly and allows researchers a low barrier of entry into using our cyberinfrastructure thereby contributing to the advancement of data and computationally intensive research in economics at the Federal Reserve Bank of Kansas City.

References

Atkins, D., Droegemeier, K.K., Feldman, S.F., Garcia-Molina, H. Klein, M.L., Messerschmitt, D.G., . . . Wright, M. H. (2003). Revolutionizing science and engineering through cyberinfrastructure: Report of the National Science Foundation blue-ribbon advisory panel on cyberinfrastructure. Retrieved from https://www.nsf.gov/cise/sci/reports/atkins.pdf

Educause. (2018). "Cyberinfrastructure." Retrieved from https://library.educause.edu/topics/infrastructure-and-emerging-technologies/cyberinfrastructure

Indiana University: Knowledge Base. (2017). "Archived: What is Cyberinfrastructure?" Retrieved from https://kb.iu.edu/d/auhf